National Research Journal of Information Technology & Information Science
Volume No: 13, (January) Year: 2026 (Special Issue)
PP: 980-988
ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

LINEARISING PARTIAL POISSON'S EQUATION USING SPARSE IDENTIFICATION NON-LINEAR DYNAMICS

Aman Bhandari

Department of CSE, Amrita Vishwa Vidyapeetham, Chennai, India

Tandra Debarati Shome

Department of CSE, Amrita Vishwa Vidyapeetham, Chennai, India

Soumyendra Singh

Department of Mathematics, Amrita Vishwa Vidyapeetham, Chennai, India

ABSTRACT

The Poisson's equation, a key PDE in modeling phenomena like heat distribution and electrostatics, is challenging to solve due to complex nonlinear terms. This paper proposes using Sparse Identification of Nonlinear Dynamics to simplify and solve Poisson's equation. SINDy extracts a sparse, simplified set of equations from observational data, potentially linearizing the nonlinear equation. The method, utilizing Lasso regression and Runge-Kutta for numerical integration, is compared to traditional PDE-solving approaches. The goal is to offer a data-driven framework to enhance the predictive capabilities of Poisson's equation, expanding its applications in real-world systems like heat conduction and electrostatics.

Keywords: Partial Poisson's Equation, Sparse Identification of Nonlinear Dynamics (SINDy), Nonlinear Dynamics, Linearization, Data-Driven Methods, Predictive Modeling

I.INTRODUCTION

The study of partial differential equations (PDEs) is fundamental to understanding a wide variety of physical phenomena, from heat diffusion to fluid dynamics. In this context, the Poisson's equation sticks out as a key PDE, shooting the distribution of scalar fields in a given area. The inherent nonlinearities present in the Poisson's equation pose demanding situations in analytical answers and necessitate modern strategies for analysis and manipulation.

This exploration delves into the interesting prospect of linearizing the Partial Poisson's Equation through the application of SINDy. by using SINDy, we aim to pick out a parsimonious set of governing phrases from observational information, in the long run looking for a linearized shape of the Poisson's equation. SINDy is a effective information-driven method for extracting the underlying dynamics of complex systems, regularly represented by way of ordinary differential equations. This technique not only provides a method to recognise the machine's behaviour in a simplified way but additionally opens avenues for superior predictive modeling and manage.

Within the next sections, we will problematic on the theoretical basis of the Poisson's equation, talk the intricacies of SINDy as a way, and delineate the steps concerned inside the manner of linearizing the Partial Poisson's Equation. via this exploration, we aspire to contribute to the advancement of strategies that bridge the theoretical complexities of partial differential equations with the realistic insights derived from information-driven methodologies.

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

II. MOTIVATION AND OBJECTIVES

The Partial Poisson's Equation is a cornerstone within the expertise of diverse bodily phenomena, playing an essential role in fields which includes heat conduction, electrostatics, and fluid dynamics. but the inherent nonlinearities within the equation regularly hinder truthful analytical answers. This motivates the exploration of alternative methodologies to unravel and simplify the complexities embedded within the Partial Poisson's Equation.

Sparse identification of Nonlinear Dynamics has completed exquisite fulfillment in unveiling the governing dynamics of complex structures from observational records. This motivates to leverage the ability of SINDy to decipher the underlying structure of the Partial Poisson's Equation, doubtlessly presenting a linearized illustration. this sort of linearization may want to offer a greater available and interpretable framework for information the gadget's behavior, with implications for stepped forward modeling and predictive abilities with the following goals:

1. Apply SINDy to Extract Dynamics: Make use of SINDy as a records-pushed tool to discover and extract the underlying dynamics embedded in observational data related to the Partial Poisson's Equation.
2. Identify Dominant Terms: Appoint sparse regression strategies to determine the dominant phrases governing the gadget's behavior, aiming to focus on the maximum influential components inside the Partial Poisson's Equation.
3. Formulate Linearized Representation: Systematically remove or linearize identified nonlinear phrases to derive a simplified, linearized illustration of the Partial Poisson's Equation.
4. Validate Against Observational Data: Validate the linearized model in opposition to extra observational data to evaluate its accuracy and generalizability, ensuring that the simplified representation captures the crucial capabilities of the gadget.
5. Explore Practical Implications: Investigate the practical implications of the linearized Partial Poisson's Equation, such as its potential applications in predictive modeling, control strategies, and facilitating a deeper understanding of the system's behavior.

Via pursuing those goals, we purpose to contribute to the improvement of methodologies that bridge the space among the theoretical intricacies of partial differential equations and the realistic insights derived from statistics-pushed strategies, with a specific consciousness on the Partial Poisson's Equation.

III. METHODOLOGY

1. Poisson's Equation

This equation can be interpreted as a Poisson equation, where the Laplacian of the scalar subject ψ is identical factor on right-hand side. The equation in the shape gives a mathematical model for the spatial distribution of ψ, inspired by means of the spatial variations of x and y.

$$\nabla^2\Psi(x,y) = e^{-x}(x - 2 + y^3 + 6y) \qquad (1)$$

This is a partial differential equation (PDE) in two variables, x and y involving the Laplace operator (∇^2) applied to a function (Ψ) [7]. Physically, the Laplacian measures the rate at which the value of (Ψ) changes around a point in space. For example, in electrostatics, (Ψ) might represent an electric potential, and $\nabla^2\Psi$ would then indicate the spatial density of

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

electric charges.

The exponential factor e^{-x}shows that the supply term decays as x will increase. because of this for larger values of x, the influence of the supply term diminishes. The polynomial equation in x and y provides shape to the supply time period, growing variations across the spatial dimensions. The Poisson equation with a non-zero supply time period implies that Ψ will not be uniform; rather, it's going to range in response to the spatial shape of $f(x, y)$. this alteration is due to the non-homogeneous nature of the source time period, which affects Ψ differently inside the $x - y$ plane.

This Poisson equation with a spatially various supply term has packages in fields where a scalar discipline is prompted with the aid of each exponential decay in a single variable and polynomial boom in another, which include in thermal diffusion, fluid go with the flow, or electromagnetism.

To clear up this Partial Differential Equation (PDE), analytical solutions may be difficult because of the nonlinearity in y, mainly the y^3 time period. Normally numerical methods like as finite distinction or finite element strategies can be used to approximate (Ψ) over a discrete grid of (x) and (y) values.

2. Boundary Conditions

These boundary conditions represent the answer's prescribed values or behaviors at the spatial domain's respective barriers. Boundary situations are important in the formula of Partial Differential Equations (PDEs) as they assist determine a unique approach to the equation. the solution to a PDE won't be particular without boundary, or it'd fail to be bodily significant. Boundary conditions outline how the solution behaves at the rims of the domain, presenting vital information on the system's behavior under particular bodily constraints.

$$\Psi(0, y) = y^3, \quad \Psi(1, y) = e^{-1}(1 + y^3) \qquad (2)$$

$$\Psi(x, 0) = xe^{-x}, \quad \Psi(x, 1) = e^{-x}(x + 1) \qquad (3)$$

The reason of specifying boundary conditions is to make certain that the solution to the PDE is nicely-posed and physically meaningful. In many cases, the choice of boundary situations displays the physical constraints or characteristics of the system under attention. as an example, in fluid dynamics, boundary conditions frequently represent the speed or strain on the boundary of the fluid area, while in warmth switch troubles, boundary situations can constitute temperature or thermal flux at the boundaries. those situations make certain the accuracy and relevance of the answer, as they assist version the bodily gadget in question.

Boundary conditions can commonly be classified into 3 sorts: Dirichlet, Neumann, and Robin situations. Dirichlet situations specify the value of the feature at the boundary, as in equations (2) and (3), wherein the characteristic (Ψ) takes regarded values on the barriers. Neumann equation, specify the spinoff of the function at the boundary, that could represent the flux or gradient of a bodily quantity like temperature or awareness. Robin boundary conditions integrate both Dirichlet and Neumann situations and are generally used whilst a combination of feature values and fluxes are recognized, inclusive of in heat transfer troubles with convective limitations. The form of boundary conditions selected relies upon at the physical nature of the hassle, and a cautious choice guarantees that the hassle is solved within the context of actual-international scenarios.

Inside the context of numerical simulations and computational techniques, the selection of boundary conditions also influences the stableness and convergence of the solution.

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

Misguided or improperly precise boundary situations can cause divergent answers or unrealistic behaviors at the area boundaries. consequently, boundary conditions ought to not simplest replicate the bodily state of affairs correctly but additionally be formulated in a way that ensures numerical balance. For instance, in fields like computational fluid dynamics (CFD) or finite detail analysis (FEA), accurately making use of boundary conditions is essential. It guarantees that mistakes such as synthetic reflections or unrealistic effects are prevented, thereby playing an important role in accomplishing reliable and meaningful results.

3. Numerical Method – Runge-Kutta

The technique is used for solving differential equations, specially powerful in scenarios requiring excessive numerical accuracy, collectively with the discretization of Partial Differential Equations (PDEs) [11]. This iterative technique approximates solutions to everyday differential equations (ODEs) of the shape $\frac{dy}{dx} = f(x, y)$ by using strategically comparing the feature $f(x, y)$ via leveraging those intermediate opinions, the Runge-Kutta approach reduces numerical errors in comparison to less complicated strategies like Euler's technique at the same time as keeping computational efficiency.

Fourth-Order Runge-Kutta (RK4) computes the price (y_{n+1}) at $(x_{n+1} = x_n + h)$, beginning from an acknowledged cost (y_n) at (x_n). The technique achieves fourth-order accuracy by combining a couple of critiques of $f(x, y)$ at intermediate points between $[x_n, x_{n+1}]$. The equation components are given by:

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \qquad (4)$$

$$The\ intermediate\ terms(k_1, k_2, k_3, k_4) are\ as\ follows:$$

$$k_1 = hf(x_n, y_n) \qquad (5)$$

$$k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right) \qquad (6)$$

$$k_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right) \qquad (7)$$

$$k_4 = hf(x_n + h, y_n + k_3) \qquad (8)$$

The Runge-Kutta method numerically solves a partial differential equation with given preliminary and boundary conditions [12]. set up a higher-decision grid for Runge-Kutta integration. Initialize solution grids for Runge-Kutta with higher decision. The results are then compared with answers acquired through Sparse identification of Nonlinear Dynamics and Lasso regression strategies [13]. This technique iteratively updates the solution matrix depending upon the given partial differential equation and boundary situations[14].

Inside the context of PDEs, the Runge-Kutta technique is specifically beneficial whilst dealing with nonlinear systems and complicated boundary conditions, because it gives a robust technique for integrating over massive domain names with varying spatial and temporal grids. As an instance, in problems related to warmth switch, fluid dynamics, or wave propagation, in which the machine's behavior is especially nonlinear and sensitive to boundary conditions, the Runge-Kutta method ensures that those elements are correctly captured over every integration step. By employing a finer grid and the use of better-order strategies like RK4, the numerical solution will become more and more specific, making this approach well-ideal for simulating physical phenomena governed by using PDEs underneath specific boundary conditions, as visible in various engineering and scientific packages.

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

4. Lasso Regression

Lasso, brief for Least Absolute Shrinkage and Selection Operator, is a regression approach that introduces a regularization term to the version. This time period encourages sparsity with the aid of reducing a number of the coefficients to 0, thereby simplifying the model and improving function choice. This penalty encourages the reduction of a few coefficients to zero, effectively selecting most effective the most crucial variables [15]. The suggest Squared mistakes (MSE) is given via the expression ($MSE = \frac{1}{n}\sum_{i=1}^{n} \quad (y_i - \hat{y_i})^2$)

Lasso regression adjusts the mean Squared error (MSE) objective with the aid of incorporating a regularization time period that penalizes the sum of the absolute values of the model's coefficients. This changed characteristic is represented as $(J(\theta) = MSE + \lambda\sum_{j=1}^{n} \quad |\theta_j|)$, where λ is the regularization parameter, and (θ_j) refers back to the model's coefficients. By using minimizing this combined goal, Lasso regression achieves a balance among becoming the information properly and preventing overfitting by enforcing sparsity in the coefficients [16].

This system proves in particular effective in figuring out a minimal set of coefficients that define the underlying machine's behavior, that's frequently defined through a partial differential equation (PDE). through the optimization system, Lasso regression helps assign significant weights to relevant spatial coordinates, dropping light on the structure of the governing PDE at the same time as simultaneously decreasing useless complexity in the model.

Within the context of solving partial differential equations (PDEs), Lasso regression can be implemented to discover key phrases or components in the discretized equations that have the maximum widespread effect on the answer. With the aid of applying Lasso to the coefficients of spatial and temporal grid points, it successfully selects the maximum influential variables, getting rid of noise or less relevant factors. This capability is especially valuable while dealing with excessive-dimensional records or complex boundary conditions, where the solution space can be big. In such cases, Lasso regression no longer only enables to streamline the model but additionally enhances interpretability by focusing on the most applicable elements of the system beneath consideration.

5. Data-Driven Method – SINDy (Sparse Identification of Nonlinear Dynamics)

This framework offers an information-centric technique designed to pick out concise and interpretable models for dynamical systems based on time-series observations [17]. This technique addresses the assignment of fixing dynamical system equations with the aid of formulating it as a sparse regression problem, stimulated by compressed sensing techniques. in lots of physical systems, only some dominant elements govern the conduct, that means the governing equations can be represented sparsely in a complex, nonlinear characteristic space. With latest advances in compressed sensing, it has become possible to correctly identify these key phrases, warding off exhaustive searches. This sparse illustration moves a balance among complexity and accuracy, helping save you overfitting.

Let $x(t) \in R^n$ represent the state of system at a given time (t), and $f\big(x(t)\big)$ denote the governing dynamics nature. The objective is to find a concise mathematical representation of (f), where only a small subset of candidate functions significantly influences the dynamics. The challenge is to efficiently select the influential terms without considering all possible combinations exhaustively.

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

The SINDy framework begins with the collection of time-series data corresponding to the state variables, $x(t)$ and $y(t)$, recorded at discrete time intervals $(t_1, t_2, \dots, t_m)$. These measurements are organized into matrices X and :

$$X=\begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_n^{(1)} & x_1^{(2)} & x_2^{(2)} & \cdots & x_n^{(2)} & x_1^{(m)} & x_2^{(m)} & \cdots & x_n^{(m)} \end{bmatrix},$$

$$Y=\begin{bmatrix} y_1^{(1)} & y_2^{(1)} & \cdots & y_n^{(1)} & y_1^{(2)} & y_2^{(2)} & \cdots & y_n^{(2)} & y_1^{(m)} & y_2^{(m)} & \cdots & y_n^{(m)} \end{bmatrix}$$

Here, $x_i^{(j)}$ and $y_i^{(j)}$ represents the i-th state at the j-th time step. These matrices encapsulate the system's time-evolving behavior for m discrete measurements. To analyze the system's evolution, functioning of the state variables in correspondence to time, $\dot{x}(t)$ and $\dot{y}(t)$, are estimated either through analytical methods or numerical differentiation. These derivative estimates form derivative matrices $(\dot{X})$ and $(\dot{Y})$:

$$\dot{X} = \begin{bmatrix} \dot{x}_1^{(1)} & \dot{x}_2^{(1)} & \cdots & \dot{x}_n^{(1)} & \dot{x}_1^{(2)} & \dot{x}_2^{(2)} & \cdots & \dot{x}_n^{(2)} & \dot{x}_1^{(m)} & \dot{x}_2^{(m)} & \cdots & \dot{x}_n^{(m)} \end{bmatrix},$$

$$\dot{Y}=\begin{bmatrix} \dot{y}_1^{(1)} & \dot{y}_2^{(1)} & \cdots & \dot{y}_n^{(1)} & \dot{y}_1^{(2)} & \dot{y}_2^{(2)} & \cdots & \dot{y}_n^{(2)} & \dot{y}_1^{(m)} & \dot{y}_2^{(m)} & \cdots & \dot{y}_n^{(m)} \end{bmatrix}$$

The matrices (X) and (Y) serve as the foundation for building a library of potential functions that represent the system's dynamics. This library, denoted as $\Theta(X, Y)$, comprises a collection of nonlinear functions. These functions include, but are not limited to, polynomials, trigonometric expressions, and other basis functions capable of capturing the intricate nonlinear behavior of the system. A typical structure for such a library might appear as follows:

$$\Theta(X,Y)=\begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & x_1y_1 & y_1^2 & sin\, sin\,(x_1) & cos\, cos\,(y_1) & \dots & \vdots\vdots\vdots\vdots\vdots\vdots\vdots\vdots\vdots\vdots \end{bmatrix}$$

The next step is to analyze regression problem to identify the coefficients of these functions. The system's evolution is modeled as:

$$\dot{X} = \Theta(X, Y)\Xi \qquad (12)$$

and similarly, for the y-component:

$$\dot{Y} = \Theta(X, Y)\Xi \qquad (13)$$

Here, (Ξ) is a sparse matrix of coefficients, where each element represents the contribution of a particular term in the library to the system's dynamics. The sparsity of (Ξ) implies that very few parameters from the library contribute to the dynamics.

Once (Ξ) is determined, the equations can be written as:

$$\dot{x_k} = f_k(x) = \Theta(x^T)\xi_k \qquad (14)$$

where $\Theta(x^T)$ is a function vector, and $(\xi_k)$is the sparse coefficient vector for the k-th equation.

Thus, the full system's dynamics can be expressed as:

$$\dot{x} = f(x) = \Xi^T\left(\Theta(x^T)\right)^T \qquad (15)$$

In practice, only X is available, and $(\dot{X})$ must be approximated. This introduces noise, resulting in:

$$\dot{X} = \Theta(X)\Xi + \eta Z \qquad (16)$$

where Z is Gaussian noise matrix, and (η) represents the noise magnitude. This overdetermined system requires a sparse solution.

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

An alternative method to LASSO for efficiently solving this problem is sequential threshold least-squares, which is less computationally expensive for large datasets.

For (y), where ($y = \dot{x}$), the equations become:

$$y_k = f_k(x) = \Theta(x^T)\xi_k \quad (17)$$

and

$$y = f(x) = \Xi^T\left(\Theta(x^T)\right)^T \quad (18)$$

In the noisy system:

$$y = \Theta(X)\Xi + \eta Z \quad (19)$$

By solving this sparse regression problem, SINDy identifies the governing equations of motion. These equations give ideas having a small number of parameter in each equation, which simplifies the modeling of complex dynamical systems from experimental data.

Preprocessing the data to mitigate noise is crucial for improving accuracy. Techniques like total variation regularization and optimal hard thresholding on singular values can help reduce noise. For high-dimensional systems such as fluid dynamics, constructing Θ directly may be impractical due to the combinatorial growth of candidate terms. The success of the SINDy approach depends on selecting appropriate measurement variables, constructing an effective library $\Theta(X)$, and implementing robust noise-handling methods. The right choice of coordinates and function basis is crucial for accurately capturing the system's dynamics with minimal terms. When prior knowledge is limited, techniques like time-delay embedding can help reconstruct underlying dynamics from a single time series.

In summary, SINDy offers a powerful and efficient method for identifying the governing equations of nonlinear systems from data. By leveraging sparsity and appropriate regression techniques, this approach provides interpretable and computationally efficient models of complex dynamical systems, even in the presence of noise.

6. Results

The 3D visualizations in Figure 1 showcase the solutions obtained using different methods: Runge-Kutta, Lasso regression, and SINDy. The Runge-Kutta solution provides a baseline, representing the traditional numerical method for solving the system. Lasso regression captures the dynamics of the partial differential equations (PDEs) through the identification of the relevant coefficients. In contrast, SINDy presents an alternative data-driven approach, where sparse identified coefficients are used to model the system's behavior.

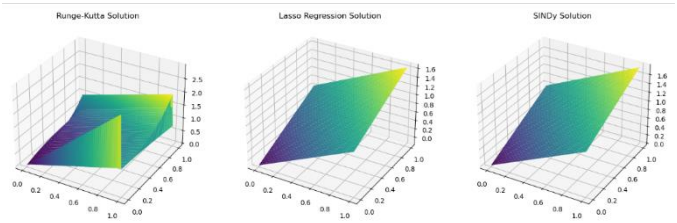


Fig. 1. Runge-kutta Solution

The 2D visualizations in Figure 2 further highlight the differences between the numerical and regression solutions. The color maps indicate the magnitude of the solutions, offering insights into the accuracy of each method.

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

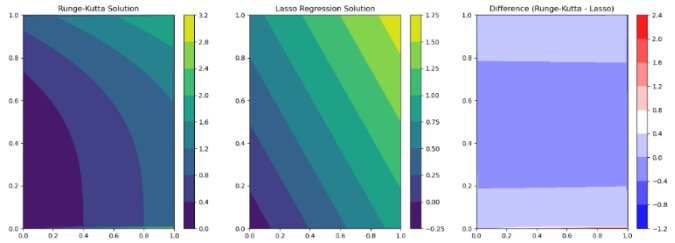


Fig. 2. SINDy Solution

CONCLUSION

In this mathematical study, we used SINDy to linearize the Poisson Equation, a fundamental partial differential equation (PDE). Our goal was to simplify the analysis of complex nonlinear systems. Using SINDy, we were able to find sparse representations of the system's basic dynamics, transforming the non-linear PDE into a linear form. To achieve this, we defined the Poisson equation with specific boundary conditions.

We employed the Runge-Kutta numerical approach to discretize the PDE. Additionally, we used Lasso regression to identify sparse coefficients representing the PDE. We then compared these results with those obtained through SINDy.

The visualizations in Figures 1 and 2 show that the Runge-Kutta method provides a baseline solution. However, we can gain deeper insights into the system dynamics using either Lasso regression or SINDy. The color maps in Figure 2 illustrate the differences between the numerical and regression solutions.

In conclusion, SINDy proves to be a valuable tool for linearizing complex PDEs and providing insights into the fundamental dynamics of nonlinear systems.

REFERENCES

1. Shea, Daniel E., Steven L. Brunton, and J. Nathan Kutz. "SINDy-BVP: Sparse identification of nonlinear dynamics for boundary value problems." Physical Review Research 3.2 (2021): 023255
2. Lee, Kookjin, Nathaniel Trask, and Panos Stinis. "Structure-preserving sparse identification of nonlinear dynamics for data-driven modeling." Mathematical and Scientific Machine Learning. PMLR, 2022.
3. Lu, Yisha, et al. "Sparse identification of nonlinear dynamical systems via non-convex penalty least squares." Chaos: An Interdisciplinary Journal of Nonlinear Science 32.2 (2022).
4. Cortiella, Alexandre, Kwang-Chun Park, and Alireza Doostan. "Sparse identification of nonlinear dynamical systems via reweighted (l_1) −regularized least squares." Computer Methods in Applied Mechanics and Engineering 376 (2021): 113620.
5. Brunton, Steven L., Joshua L. Proctor, and J. Nathan Kutz. "Discovering governing equations from data by sparse identification of nonlinear dynamical systems." Proceedings of the national academy of sciences 113.15 (2016): 3932-3937.
6. Messenger, Daniel A., and David M. Bortz. "Weak SINDy: Galerkin-based data-driven model selection." Multiscale Modeling & Simulation 19.3 (2021): 1474-1497.
7. Rudy, Samuel H., et al. "Data-driven discovery of partial differential equations." Science advances 3.4 (2017): e1602614.
8. Champion, Kathleen, et al. "Data-driven discovery of coordinates and governing equations." Proceedings of the National Academy of Sciences 116.45 (2019): 22445-22451.
9. Schaeffer, Hayden. "Learning partial differential equations via data discovery and sparse optimization." Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 473.2197 (2017): 20160446.

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

10. Loiseau, Jean-Christophe, and Steven L. Brunton. "Constrained sparse Galerkin regression." Journal of Fluid Mechanics 838 (2018): 42-67.

11. Zhang, Linan, and Hayden Schaeffer. "On the convergence of the SINDy algorithm." Multiscale Modeling \& Simulation 17.3 (2019): 948-972.

12. Mangan, Niall M., et al. "Inferring biological networks by sparse identification of nonlinear dynamics." IEEE Transactions on Molecular, Biological, and Multi-Scale Communications 2.1 (2016): 52-63.

13. Kaiser, Eurika, J. Nathan Kutz, and Steven L. Brunton. "Sparse identification of nonlinear dynamics for model predictive control in the low-data limit." Proceedings of the Royal Society A 474.2219 (2018): 20180335.

14. Roy, Bappaditya, and Natsuhiko Yoshinaga. "Estimation of spatial and time scales of collective behaviors of active matters through learning hydrodynamic equations from particle dynamics." arXiv preprint arXiv:2411.03783 (2024).

15. Fukami, Kai, et al. "Sparse identification of nonlinear dynamics with low-dimensionalized flow representations." Journal of Fluid Mechanics 926 (2021): A10.

16. Tran, Giang, and Rachel Ward. "Exact recovery of chaotic systems from highly corrupted data." Multiscale Modeling & Simulation 15.3 (2017): 1108-1129.

17. Schaeffer, Hayden, Giang Tran, and Rachel Ward. "Extracting sparse high-dimensional dynamics from limited data." SIAM Journal on Applied Mathematics 78.6 (2018): 3279-3295.

18. Rudy, Samuel, et al. "Data-driven identification of parametric partial differential equations." SIAM Journal on Applied Dynamical Systems 18.2 (2019): 643-660.

19. Brunton, Steven L., and J. Nathan Kutz. "7 Data-driven methods for reduced-order modeling." Model Order Reduction 2 (7): 307-344.

20. Champion, Kathleen, et al. "A unified sparse optimization framework to learn parsimonious physics-informed models from data." IEEE Access 8 (2020): 169259-169271.